

# SYSTEM AND METHOD FOR IMAGE SENSING AND PROCESSING

## SPECIFICATION

### BACKGROUND OF THE INVENTION

5 A number of important compression standards for still and video images employ the discrete cosine transform (DCT). For example, Fig. 1 illustrates a standard JPEG algorithm for compressing a still image. In the illustrated algorithm, the image is divided into 8x8 pixel blocks of pixel intensity values (e.g., illustrated block 102). For each 8x8 block 102, the two-dimensional (2-D) DCT is computed  
10 (step 104). The DCT coefficients are scaled, quantized, and truncated (i.e., rounded off) (step 106) to retain only the information that is most important for accurate perception by the human eye. For example, because the eye is relatively insensitive to high spatial frequencies, and because the largest DCT coefficients are typically those representing the lowest spatial frequencies, many of the high-frequency DCT  
15 coefficients can be rounded to zero in the quantization step 106. The quantized coefficients are then entropy encoded — typically using Huffman encoding — for more compact representation the remaining, non-zero DCT coefficients (step 108). The above-described compression scheme can, for example, be applied separately to different spectral components of a color image — e.g., the red, green and blue pixels in  
20 an RGB image or the luminance-chrominance values of the image. Because the DCT is a linear operation it can be applied separately to any linear combination of RGB pixel values.

The 2-D, NxN point DCT is defined as follows,

$$DCT\{A\}(k,l) = \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \alpha(k) \cdot \alpha(l) \cdot A \left[ \left( n + \frac{1}{2} \right) T, \left( m + \frac{1}{2} \right) T \right] \cdot \cos \left( \frac{\pi \cdot k \cdot (2n+1)}{2N} \right) \cdot \cos \left( \frac{\pi \cdot l \cdot (2m+1)}{2N} \right)$$

25 (1a)

where:  $\alpha(0) = \sqrt{\frac{1}{N}}, \alpha(k) = \sqrt{\frac{2}{N}}, k = 1, 2, 3, \dots, N-1$  , (1b)

and where  $A$  denotes the sampled image,  $n$  and  $m$  denote the spatial sampling indices, and  $k$  and  $l$  denote the spatial frequency indices. Computation of the 8x8 DCT can require on the order of  $(2 \times 8 \times 8) \times (8 \times 8) = 8192$  multiplications, although some well known algorithms are capable of reducing the number of multiplications by a factor of 50 or more. Nonetheless, computation of the DCT typically comprises the bulk of the computations required for image compression. Furthermore, although some compression technologies — such as JPEG2000 — use wavelet representations rather than the DCT, DCT-based technologies are expected to remain in widespread use for the foreseeable future.

Moreover, in addition to the JPEG standard, which is used for still image compression, there are a number of commonly used video compression standards — e.g., Motion JPEG, MPEG(1,2,4), and H.26X — which require computation of the DCT of each frame of the video frame sequence.

Currently, in most commercial applications, image compression is performed by separate digital signal processing circuits which derive DCT coefficients based on digitized image data. However, conventional DCT algorithms require a substantial amount of computing power and consume a large amount of power, which makes such image processing less attractive for devices in which power conservation is important. Such devices include, for example, mobile camera phones, digital cameras, and wireless image sensors for machine health monitoring and surveillance.

### SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide an image sensing and processing system which reduces the number of computations, particularly multiplications, required to derive DCT coefficients from image data.

It is a further object of the present invention to provide such a system which reduces the amount of power consumed by the derivation of DCT coefficients.

These and other objects are accomplished by a system which computes DCT coefficients of an image using the Arithmetic Fourier Transform (AFT). The AFT method enables computation of the Fourier transform primarily by performing additions. Other than pre-scaling of the pixel data, no multiplication is required. In  
5 hardware realizations, the greater computational efficiency of the AFT allows savings in circuit complexity, size, and power consumption, and also increases processing speed. The image is preferably sampled using non-uniformly spaced sensors, although non-uniform sampling can also be achieved by interpolation of signals from a set of uniformly spaced sensors. The AFT algorithm can be implemented in either  
10 digital or analog circuitry. The AFT techniques of the present invention, particularly the analog implementations, allow vast economies in circuit complexity and power consumption.

In accordance with one aspect of the present invention, incoming light is detected by a sensor array comprising at least first and second sensors having first  
15 and second sensor locations, respectively. The first sensor location is proximate to a location of a first extremum of a basis function of a domain transform, a basis function having one or more spatial coordinates defined according to the spatial coordinate system of the sensor array. The second sensor location is proximate to a location of a second extremum of the same basis function or a different basis function.  
20 The system includes at least one filter which receives signals from the first and second sensors and generates a filtered signal comprising a weighted sum of at least the signals from the first and second sensors. We include the special case of the foregoing in which the signal from a single sensor may comprise a filter output.

In accordance with additional aspect of the present invention, incoming  
25 light is detected by a sensor array comprising a plurality of sensors, including at least first and second sensors having first and second sensor locations, respectively. The incoming light signal has a first value at the first sensor location and second value at the second sensor location. The system includes an interpolation circuit which receives signals from the first and second sensors, these signals representing the first  
30 and second values, respectively, of the incoming light signal. The interpolation circuit interpolates the signals from the first and second sensors to generate an

interpolated signal. The interpolated signal represents an approximate value of the incoming light signal at a location proximate to a first extremum of at least one basis function of a domain transform, the at least one basis function having at least one spatial coordinate defined according to the spatial coordinate system of the sensor array.

### BRIEF DESCRIPTION OF THE DRAWINGS

Further objects, features, and advantages of the present invention will become apparent from the following detailed description taken in conjunction with the accompanying drawings showing illustrative embodiments of the present invention, in which:

Fig. 1 is a block diagram illustrating an exemplary prior art image processing procedure;

Fig. 2 is a diagram illustrating data processed in accordance with the present invention;

Fig. 3 is a diagram and accompanying graphs illustrating an exemplary image sampling space and corresponding domain transform basis functions in accordance with the present invention;

Fig. 4 is a graph illustrating error characteristics of an exemplary system and method for image sensing and processing in accordance with the present invention;

Fig. 5 is a diagram illustrating an exemplary image sampling space in accordance with the present invention;

Fig. 6 is a graph illustrating error characteristics of an exemplary system and method for image sensing and processing in accordance with the present invention;

Fig. 7 is a graph illustrating error characteristics of an additional exemplary system and method for image sensing and processing in accordance with the present invention;

Fig. 8 is a graph illustrating error characteristics of yet another exemplary system and method for image sensing and processing in accordance with the present invention;

Fig. 9 is a diagram illustrating an exemplary image sampling space in accordance with the present invention;

Fig. 10 is a diagram illustrating an exemplary sensor array and filter arrangement in accordance with the present invention;

Fig. 11 is a flow diagram illustrating an exemplary image sensing and processing procedure in accordance with the present invention;

Fig. 12 is a flow diagram illustrating an exemplary signal filtering procedure for use in the procedure illustrated in Fig. 11;

Fig. 13 is a flow diagram illustrating an exemplary image sensing and processing procedure in accordance with the present invention;

Fig. 14 is a flow diagram illustrating an exemplary signal filtering procedure for use in the procedure illustrated in Fig. 13;

Fig. 15 is a diagram illustrating an exemplary sensor array and filtering circuit in accordance with the present invention;

Fig. 16 is a flow diagram illustrating an exemplary image sensing and processing procedure in accordance with the present invention;

Fig. 17 is a timing diagram associated with Fig. 10, illustrating an exemplary timing sequence produced by the clock generator to generate the filtered signal S(3,12).

Fig. 18 is a diagram illustrating an exemplary sensor array and filter arrangement in accordance with the present invention;

Throughout the drawings, unless otherwise stated, the same reference numerals and characters are used to denote like features, elements, components, or portions of the illustrated embodiments.

#### DETAILED DESCRIPTION OF THE INVENTION

An incoming image signal — such as an incoming light pattern from a scene being imaged — can be sampled by an array of sensors such as a charge

coupled device (CCD). In accordance with the present invention, the individual sensors in the array can be distributed according to a spatial pattern which is particularly well suited for increasing the efficiency of AFT algorithms. The preferred spatial distribution for a 2-D sensor array can be better understood by first considering the one-dimensional (1-D) case. For example, to find the 1-D AFT that is equivalent to an 8-point, 1-D DCT on a unit interval (0 to 1) of space or time, 12 non-uniformly spaced samples should be used. The preferred sampling locations are (0, 1/4, 2/7, 1/3, 2/5, 1/2, 4/7, 2/3, 3/4, 4/5, 6/7, 1) — although it is to be noted that, if the entire signal being sampled includes multiple unit intervals, the first and the last samples of each interval are shared with any adjacent unit intervals. In number theory, fractions of the form  $k/j$ , where  $k=0,1\dots N-1$  and  $j=1,2\dots N$  are commonly referred to as "Farey fraction" of order  $N$ . It can thus be seen that the above-described sampling locations — which provide the preferred set of samples for calculating an 8-point DCT based on the corresponding, 12-point AFT — correspond to an even subset of Farey fractions of order 8 defined as  $2k/j$ , where  $k=0,1\dots 4$  and  $j=1,2\dots 8$ .

The above-described signal samples can be used, in conjunction with a function known as the Mobius function, to compute the AFT of the signal. The 1-D AFT based on the Mobius function is well known; an exemplary derivation of the transform can be found in D.W. Tufts, G. Sadasiv, "Arithmetic Fourier Transform and Adaptive Delta Modulation: a Symbiosis for High Speed Computation," *SPIE Vol. 880 High Speed Computing* (1988). The 1-D Mobius function  $\mu_1(n)$  is defined as follows:

$$\mu_1(1) = 1 \quad (2a)$$

$$\mu_1(n) = (-1)^s \text{ if } n=(p_1)(p_2)(p_3)\dots(p_s), \text{ where } p_i \text{ are different prime numbers} \quad (2b)$$

$$\mu_1(n) = 0 \text{ if } p^2|n \text{ for any prime number } p, \quad (2c)$$

where the vertical bar notation  $m|n$  means that the integer  $n$  is divisible by the integer  $m$  with no remainder. If  $n$  can be expressed as the product of  $s$  different prime numbers, the value of  $\mu_1(n)$  is  $(-1)^s$ ; otherwise, the value is zero.

Within a unit interval, the signal  $A(t)$  is assumed to be periodic with period one. If the signal  $A(t)$  is further assumed to be band-limited to a total of  $N$  harmonics, its AFT coefficients are given by:

$$a_k(t_{ref}) = \sum_{m=1}^{\infty} \mu_1(m) \cdot S(mk, t_{ref}) \quad \text{for } k = 1, 2, 3, \dots, N, \quad (3)$$

- 5 where each  $S(n, t_{ref})$  denotes the output of a filter having the following filtering function, based on samples  $A(t_{ref} - \frac{j}{n})$  which are distributed at locations corresponding to respective Farey fractions of the interval 0 to 1:

$$S(n, t_{ref}) = \frac{1}{n} \sum_{j=0}^{n-1} A(t_{ref} - \frac{j}{n}) \quad \text{for } n = 1, 2, 3, \dots, N. \quad (4)$$

- Each of the filter outputs  $S(n, t_{ref})$  is the sum of the respective samples  $A(t_{ref} - \frac{j}{n})$   
 10 multiplied by the scale factor  $1/n$ , where  $t_{ref}$  is an arbitrary reference time.  $t_{ref}$  is preferably equal to 1 for a unit interval. Each AFT coefficient is the sum of the filter outputs of selected filters, weighted by the Mobius function  $\mu_1(m)$ .

- To process a 2-D input signal such as an image or an image portion (e.g., a unit sub-image or block), the AFT is extended to two dimensions using a 2-D  
 15 Mobius function  $\mu_2(n, m)$  which is defined as follows:

$$\mu_2(n, m) = \mu_1(n) \mu_1(m), \quad (5)$$

where  $n$  and  $m$  are positive integers, and  $\mu_1(n)$  is the 1-D Mobius function defined in Eqs. 2a, 2b, and 2c.

- Formulae for the 2-D AFT of a zero-mean 2-D input signal  $A(p, q)$  –  
 20 where  $p$  and  $q$  are continuous spatial coordinates in a unit range (i.e., a range from 0 to 1) – can be represented with respect to any arbitrary reference point  $(p_{ref}, q_{ref})$  by a 2-D Fourier series as follows:

$$A(p_{ref}, q_{ref}) = \sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{k,l}(p_{ref}, q_{ref}) \quad (6)$$

$$a_{k,l}(p_{ref}, q_{ref}) = A_{k,l} \cos(2\pi \cdot k p_{ref} + \theta_k) \cos(2\pi \cdot l q_{ref} + \theta_l), \quad (7)$$

where  $(p_{ref}, q_{ref})$  is an arbitrary reference location, preferably (1, 1) for a unit sub-image.

It is assumed that the signal  $A(p, q)$  is band-limited to  $N$  harmonics in both spatial dimensions  $p$  and  $q$  — i.e., the Fourier series coefficients higher than  $N$  are equal to zero. A filter-bank having  $N^2$  filters is used to process the image data, each filter having the following filtering function:

$$S(n, m, p_{ref}, q_{ref}) = \frac{1}{n} \frac{1}{m} \sum_{j=0}^{n-1} \sum_{k=0}^{m-1} A(p_{ref} - \frac{j}{n}, q_{ref} - \frac{k}{m}) \quad (8)$$

where  $n = 1, 2, \dots, N$  and  $m = 1, 2, \dots, N$ . It can be seen from Eq. (8) that the spatial locations  $(p_{ref} - \frac{j}{n}, q_{ref} - \frac{k}{m})$  of the samples  $A(p_{ref} - \frac{j}{n}, q_{ref} - \frac{k}{m})$  processed by the filters are defined — relative to the reference location  $(p_{ref}, q_{ref})$  — by respective Farey fractions  $\frac{j}{n}$  and  $\frac{k}{m}$  of the dimensions of a unit image block, as is discussed in further detail below with respect to Fig. 3.

By replacing the signal  $A(p, q)$  in Eq. (8) by its Fourier series given in Eqs. (6) and (7), it can be shown that the output of each filter is equal to the sum of a particular set of Fourier series coefficients of  $A(p, q)$  :

$$S(n, m, p_{ref}, q_{ref}) = \sum_{j=1}^{\infty} \sum_{k=1}^{\infty} a_{nl, mk}(p_{ref}, q_{ref}) = \sum_{\substack{j \text{ such that} \\ n|j}} \sum_{\substack{k \text{ such that} \\ m|k}} a_{jk}(p_{ref}, q_{ref}) \quad (9)$$

A derivation of Eq. (9) is provided in Appendix A attached hereto.

Based on the assumption that the signal is band-limited, there are no more than  $\left\lfloor \frac{N}{n} \right\rfloor + \left\lfloor \frac{N}{m} \right\rfloor$  terms that are non-zero, where  $\lfloor x \rfloor$  denotes the largest integer which is less than or equal to  $x$ . Given Eq. (9) it is possible to prove the following relation for the 2-D Fourier series coefficients (a proof is provided in Appendix B attached hereto):

$$a_{kl}(p_{ref}, q_{ref}) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu_2(m, n) \cdot S(mk, nl, p_{ref}, q_{ref}) \quad \text{for } k, l = 1, 2, \dots, N. \quad (10)$$



Furthermore, because of the close relationship between the DCT and the Discrete Fourier Transform (DFT), the above-described outputs from the 2-D AFT algorithm can be used to calculate the DCT coefficients of a unit sub-image divided into  $N \times N$  uniformly spaced pixels. First, the image sensor array is divided into unit area blocks of pixels, each block having by definition, a size of  $1 \times 1$ . The photosensitive elements inside each unit area are placed in locations based on a set of Farey fractions of the unit block size, to provide the appropriate samples for the filters defined in Eq. (8). In order to calculate the filters' outputs, an appropriate reference location  $(p_{ref}, q_{ref})$  is chosen. A convenient reference location is at  $p_{ref} = 1$  and  $q_{ref} = 1$  (at a corner of the unit area). Eq. (8) then becomes:

$$S(n, m) = \frac{1}{n} \frac{1}{m} \sum_{j=0}^{n-1} \sum_{k=0}^{m-1} A\left(1 - \frac{j}{n}, 1 - \frac{k}{m}\right) \quad (11)$$

where  $n=1, 2, \dots, N$  and  $m=1, 2, \dots, N$ .

The output of the 2-D AFT is a set of 2-D Fourier series coefficients. In order to derive DCT coefficients from the Fourier Series coefficients, an extended image block  $X(p, q)$  is derived by extending the original image block  $A(p, q)$  by its own mirror image in both directions, as shown in Fig. 2, as follows:

$$X(p, q) = \begin{cases} A(p, q) & 0 \leq p < 1, 0 \leq q < 1, \\ A(2 - p, q) & 1 \leq p < 2, 0 \leq q < 1, \\ A(p, 2 - q) & 0 \leq p < 1, 1 \leq q < 2, \\ A(2 - p, 2 - q) & 1 \leq p < 2, 1 \leq q < 2 \end{cases} \quad (12)$$

If the AFT is to be computed from extended image block  $X(p, q)$ , rather than from the original block  $A(p, q)$ , the appropriate filter values are:

$$S(n, m) = \frac{1}{n} \frac{1}{m} \sum_{j=0}^{n-1} \sum_{k=0}^{m-1} X\left(2 - \frac{2 * j}{n}, 2 - \frac{2 * k}{m}\right) . \quad (13)$$

If the extended image block  $X(p, q)$  obeys the Nyquist criterion, the resulting AFT coefficients are equal to the DCT coefficients within a scale factor — a proof of this result is provided in Appendix C attached hereto. On the other hand, if the extended image does not satisfy the Nyquist criterion, the 2-D AFT coefficients

are only an approximation of the 2-D DCT coefficients. This situation is more likely to occur for images rich in high-frequency components. However, it is possible to improve the approximation using aliasing correction techniques which are discussed in further detail below.

- 5 In any case, from Eqs. (12) and (13), the respective outputs  $S(n,m)$  of the filters can be expressed as follows:

$$S(n,m) = \frac{1}{n} \frac{1}{m} \cdot \left[ \sum_{j=0}^{\left\lfloor \frac{n}{2} \right\rfloor} \sum_{k=0}^{\left\lfloor \frac{m}{2} \right\rfloor} A\left(\frac{2j}{n}, \frac{2k}{m}\right) + \sum_{j=0}^{\left\lfloor \frac{n}{2} \right\rfloor} \sum_{k=1}^{\left\lceil \frac{m}{2} \right\rceil - 1} A\left(\frac{2j}{n}, \frac{2k}{m}\right) + \dots \right. \\ \left. \sum_{j=1}^{\left\lceil \frac{n}{2} \right\rceil - 1} \sum_{k=0}^{\left\lfloor \frac{m}{2} \right\rfloor} A\left(\frac{2j}{n}, \frac{2k}{m}\right) + \sum_{j=1}^{\left\lceil \frac{n}{2} \right\rceil - 1} \sum_{k=1}^{\left\lceil \frac{m}{2} \right\rceil - 1} A\left(\frac{2j}{n}, \frac{2k}{m}\right) \right], \quad (14)$$

where  $n$  and  $m$  take the values from 1 to  $N$ .  $\lceil x \rceil$  denotes the smallest integer which is greater than or equal to  $x$ .

- 10 From Eq. 14 it is apparent that there are certain points in the sample space that are repeated. As a result, by calculating the DCT rather than the DFT, the number of independent points in the 2-D AFT is decreased by nearly one-half. For example, to calculate an 8x8 point DCT inside the unit sub-image, a set of 12x12 photosensitive elements per unit area is used. The elements at the edges of the unit
- 15 area are shared between adjacent sub-images, thus reducing the effective number of points per block to 11x11. An exemplary non-uniform sample space 300 is illustrated in Fig. 3. In the illustrated example, non-uniformly distributed sample points 348 are used for the 2-D AFT calculation. The corresponding effective DCT sample points 398 are distributed uniformly.

- 20 With the image sampled as illustrated in Fig. 3, and using filters whose filtering functions are defined according to Eq. (14), the 2-D AFT coefficients  $x_{k,l}$  can be computed as follows:

$$x_{k,l} = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu_2(m,n) \cdot S(mk, nl) \quad \text{for } k, l = 1, 2, \dots, N \quad (15a)$$

$$x_{k,0} = \sum_{m=1}^{\infty} \mu_1(m) \cdot S(mk, N) \quad \text{for } k = 1, 2, \dots, N \quad (15b)$$

$$x_{0,l} = \sum_{n=1}^{\infty} \mu_1(n) \cdot S(N, nl) \quad \text{for } l = 1, 2, \dots, N \quad (15c)$$

$$x_{0,0} = E[A], \quad (15d)$$

where  $E[A]$  is the mean value of the image,  $x_{k,l}$  are the 2-D AFT coefficients of the extended block image  $X$ ,  $x_{k,0}$  are the coefficients obtained by calculating the 1-D AFT of the mean values of the rows along the p-axis, and  $x_{0,l}$  are the coefficients obtained by calculating the 1-D AFT of the mean values of the columns along the q-axis.

The corresponding DCT coefficients can be computed as follows:

$$DCT\{A\}(0,0) = 8 \cdot E[A] \quad (15e)$$

$$10 \quad DCT\{A\}(k,0) = 4\sqrt{2} \cdot x_{k,0} \quad k=1, 2, \dots, N-1 \quad (15f)$$

$$DCT\{A\}(0,l) = 4\sqrt{2} \cdot x_{0,l} \quad l=1, 2, \dots, N-1 \quad (15g)$$

$$DCT\{A\}(k,l) = 4 \cdot x_{k,l} \quad k=1, 2, \dots, N-1 \text{ and } l=1, 2, \dots, N-1 \quad (15h)$$

The above discussion demonstrates that using the 2-D AFT to compute the DCT coefficients of an image portion allows the entire computation to be performed primarily with addition operations, and with very few multiplication operations, thus making the 2-D AFT procedure extremely efficient. The source of this increased efficiency can be further understood with reference to Fig. 3. The drawing illustrates an exemplary 2-D sample area 300 of a sensor array corresponding to the area of a conventional 8 x 8 block of pixels 398 arranged in a conventional pattern. However, in accordance with the present invention, the illustrated region 300 has certain preferred locations 348 for use with the above-described 2-D AFT technique. The preferred locations 348 correspond to extrema (i.e., maxima) of basis functions of the transform being performed. For example, it is well known that the

basis functions of a Fourier transform are sine and cosine functions of various different frequencies (in the case of a time-varying signal) or wavelengths (in the case of a spatially varying signal such as an image). In the case of a cosine transform such as a DCT, the basis functions are cosine functions of various frequencies (for time-varying signals) or wavelengths (for spatially varying signals), as given by Eq. 1. In the exemplary sample area 300 illustrated in Fig. 3, columns 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, and 342 correspond to the locations of respective maxima 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, and 312 of cosine basis functions 320, 321, 322, 323, 324, 325, 326, and 327, where the spatial coordinate  $q$  of these basis functions is defined according to the spatial coordinate system of either the sensor array or the illustrated region 300. In particular, in the illustrated example, the spatial coordinate  $q$  of the aforementioned basis functions 320, 321, 322, 323, 324, 325, 326, and 327 is equal to the horizontal coordinate of the sensor array, referenced to the left edge (column 331) of the illustrated region 300. Similarly, rows 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, and 392 of the preferred sample locations 348 correspond to respective extrema 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, and 362 of cosine basis functions 370, 371, 372, 373, 374, 375, 376, and 377, these basis functions having a vertical spatial coordinate  $p$  which, similarly to  $q$ , is defined according to the spatial coordinate system of either the sensor array or the illustrated region 300 thereof.

The 2-D AFT calculation uses only selected samples such that, for each selected sample, the relevant basis function has a value of +1 at the location of the sample. Such a sampling pattern allows the simplifying assumption that, when computing the AFT coefficients  $x_{k,l}$ , the pre-scaled input sensor signals need only be multiplied by a factor of +1, 0 or -1 — hence the use of the 2-D Mobius function  $\mu_2(m,n)$  in Eq. (10).

Fig. 10 illustrates an exemplary portion 1004 of a sensor array 1034, along with a filter arrangement 1022 for detecting an incoming signal (e.g., a light pattern being received from a scene being imaged) and processing the signal to derive the respective filter outputs  $S(n,m)$  in Eq. (14). The sensor array portion 1004 has sensors 1002 located in the preferred locations for the AFT calculation, these

locations being defined to have vertical and horizontal distances, relative to corner pixel 1028, which are equal to various Farey fractions multiplied by the size 1032 of the array portion 1004. Optionally, the filtering can be performed by an analog circuit 1022 as is illustrated in Fig. 10 or by a digital filter 1502 as is illustrated in Fig. 15. In either case, column selection operations are preferably performed by a column selector 1036 under control of a microprocessor 1018, and the respective filter outputs  $S(n,m)$  are stored in a memory device such as RAM 1016.

Regardless of whether an analog filter 1022 or a digital filter 1502 is being used to compute the filter outputs  $S(n,m)$ , the illustrated arrangement can be operated according to the exemplary procedure illustrated in Fig. 11. In the illustrated procedure, an incoming signal—e.g., a light pattern from a scene—is received by the sensor array 1004 (step 1102). The incoming signal is detected by the respective sensors 1002 of the array 1004 to generate sensor signals (step 1104), and the signals are received by the analog or digital filter arrangement 1022 or 1502 (step 1106). Respective weighted sums of respective sets of sensor signals are derived to generate respective filtered signals (step 1118). For example, a weighted sum of a set of sensor signals (e.g., a weighted sum of the respective pixel values 1028, 1029, 1030, and 1031 from the intersections of the rows 1024 and 1026 with the columns 1044 and 1046) is derived by the filter 1022 or 1502 to generate a filtered signal  $S(2,3)$  (step 1118).

In the case of an analog filter arrangement 1022, the weighted sums derived in steps 1108 and 1110 can be produced in accordance with the procedure illustrated in Fig. 12. In the illustrated filtering procedure 1108 or 1110, the signals from the respective sensors are amplified with the appropriate gains to generate respective amplified signals (step 1208). For example, the signal from the first sensor 1028 in row 1024 and column 1044 is amplified with a first gain to generate a first amplified signal (step 1202), the signal from the second sensor 1029 in the row 1024 and column 1046 is amplified with a second gain to generate a second amplified signal (step 1204), etc. The resulting amplified signals are integrated to generate the filtered signal (step 1206).

The operation of the analog filtering circuit 1022 illustrated in Fig. 10 can be further understood with reference to the timing diagram illustrated in Fig. 17. First, the microprocessor 1018 determines which filter is to be calculated — i.e., selects values for  $n$  and  $m$ . Given the value  $m$ , the appropriate columns and  $\Phi_{amp}^m$  are selected. Then, given the value of  $n$ , appropriate  $\Phi_{int}^n$  and  $\Phi_{sj}^i$  are selected. An exemplary timing cycle for calculating the filter S(3,12) is as follows:

1.  $n=3, m=12,$
2.  $\Phi_{int}^3=1, \Phi_{int}^i=0,$  where  $i = 1,2,4,5,6,7,12,$
3. Select Column 0
- 10 4.  $\Phi_{s2}^1=1, \Phi_{s2}^8=1,$  other  $\Phi_{sj}^i=0$
5. Transfer charge to integrator 1010,  $\Phi_t=1, \Phi_{sj}^i=0$
6. Select Column 1/6
7.  $\Phi_{s1}^1=1, \Phi_{s1}^8=1,$  other  $\Phi_{sj}^i=0$
8. Transfer charge to integrator 1010,  $\Phi_t=1, \Phi_{sj}^i=0$
- 15 9. Select Column 1/3
10.  $\Phi_{s1}^1=1, \Phi_{s1}^8=1,$  other  $\Phi_{sj}^i=0$
11. Transfer charge to integrator 1010,  $\Phi_t=1, \Phi_{sj}^i=0$
12. Select Column 1/2
13.  $\Phi_{s1}^1=1, \Phi_{s1}^8=1,$  other  $\Phi_{sj}^i=0$
- 20 14. Transfer charge to integrator 1010,  $\Phi_t=1, \Phi_{sj}^i=0$
15. Select Column 2/3
16.  $\Phi_{s1}^1=1, \Phi_{s1}^8=1,$  other  $\Phi_{sj}^i=0$
17. Transfer charge to integrator 1010,  $\Phi_t=1, \Phi_{sj}^i=0$
18. Select Column 5/6

19.  $\Phi_{s1}^1=1$ ,  $\Phi_{s1}^8=1$ , other  $\Phi_{sj}^i=0$
20. Transfer charge to integrator 1010,  $\Phi_i=1$ ,  $\Phi_{sj}^i=0$
21. Sample the integrator's output,  $\Phi_{s3}=1$
22.  $\Phi_{amp}^{12}=1$ ,  $\Phi_{amp}^i=0$ , where  $i = 1,2,3,4,5,6,7$ ,
23. Transfer charge to amplifier 1012,  $\Phi_{s3}=0$ ,  $\Phi_{i3}=1$
24. Perform AD conversion using ADC 1014 and store the digital value  $S(3,12)$   
in RAM 1016
25. Reset the integrator 1010 and amplifier 1012

Once the respective filter outputs  $S(m,n)$  are derived, the 2-D AFT coefficients are derived (step 1112). To derive the AFT coefficients (step 1112), the filter outputs are weighted using appropriate values of the Mobius function as is described above with respect to Eqs. (15a)-(15d) above (step 1114), and the resulting weighted signals are added/summed in accordance with Eqs. (15a)-(15d) (step 1116).

It is to be noted that, if a digital filter 1502 is used, as is illustrated in Fig. 15, the respective signals from the sensors 1002 in the array 1004 are preferably amplified by amplifiers 1006, and the resulting amplified signals are then received (converted to digital values) and processed by the digital filter 1502. Those skilled in the art will be familiar with numerous commercially available, individually programmable, special-purpose digital filters which can easily be programmed by ordinarily skilled practitioners to perform the mathematical operations described above. Because the resolution of the analog-to-digital converter (ADC) 1014 in a typical image sensor system is no greater than 12 bits, a 16-bit digital signal processor is suitable for use as the digital filter 1502.

The 2-D AFT is based on the assumption that the mean intensity value (a/k/a the "DC" value) of the full sub-image, as well as mean value for each row and column separately, is zero. If there is a non-zero DC value for a row, column, or the entire sub-image, that value is preferably used to derive correction values for

adjusting the appropriate filter outputs  $S(n,m)$ . The proper correction amounts for the case when the entire sub-image has a non-zero mean  $E[A]$ , are as follows:

$$\Delta(k,0) = -E[A] \left( \sum_{m=1}^{\left\lfloor \frac{N}{k} \right\rfloor} \mu_1(m) \right) \quad k=1,2..N-1; \quad \Delta(0,l) = -E[A] \left( \sum_{m=1}^{\left\lfloor \frac{N}{l} \right\rfloor} \mu_1(m) \right) \quad l=1,2..N-1 \quad (16a)$$

and

$$5 \quad \Delta(k,l) = -E[A] \left( \sum_{m=1}^{\left\lfloor \frac{N}{k} \right\rfloor} \mu_1(m) \right) \left( \sum_{n=1}^{\left\lfloor \frac{N}{l} \right\rfloor} \mu_1(n) \right) \quad k,l=1,2..N-1 \quad (16b)$$

In addition, correction amounts should be computed if the input signal has non-zero mean values in any of the rows or columns (i.e., if  $x_{k,0}$  or  $x_{0,l}$  are non-zero). In the case of non-zero mean values in rows or columns, it is sufficient to correct only  $x_{k,l}$ , where  $k=1,2....N-1$  and  $l=1,2....N-1$ . The correction formula is as follows:

$$10 \quad \Delta_{local}(k,l) = -x_{k,0} \left( \sum_{m=1}^{\left\lfloor \frac{N}{l} \right\rfloor} \mu_1(m) \right) - x_{0,l} \left( \sum_{m=1}^{\left\lfloor \frac{N}{k} \right\rfloor} \mu_1(m) \right) \quad (17)$$

The correction factors  $\Delta(k,l)$  and  $\Delta_{local}(k,l)$  are then added to the uncorrected 2-D AFT coefficients  $x_{k,l}$  to derive corrected 2-D AFT coefficients  $A_c(k,l)$  as follows:

$$15 \quad A_c(k,l) = x_{k,l} + \Delta(k,l) \quad k=0, l=1,2....N-1 \text{ or } k=1,2...N-1, l=0 \quad (18a)$$

$$A_c(k,l) = x_{k,l} + \Delta(k,l) + \Delta_{local}(k,l) \quad k,l=1,2....N-1 \quad (18b)$$

As an illustrative example, the 8 x 8 DCT case will now be considered. It is not necessary to determine exactly the respective mean values of the entire unit-area sub-image and of the local rows and columns. Rather, it is sufficient to use estimates for these mean values. For the mean value  $E[A]$  of the entire sub-image  $A$ ,



the closest estimate, in terms of least mean-square error, is provided by the filter output that averages the largest number of points. In the general,  $N \times N$  case this is  $S(N, N)$ . In the case of an  $8 \times 8$  DCT, the best estimate of the mean  $E[A]$  of the entire sub-image  $A$  is as follows:

$$E[A] = S(8, 8) = \frac{1}{64} \left[ \sum_{j=0}^4 \sum_{k=0}^4 A\left(\frac{j}{4}, \frac{k}{4}\right) + \sum_{j=0}^4 \sum_{k=1}^3 A\left(\frac{j}{4}, \frac{k}{4}\right) + \sum_{j=1}^3 \sum_{k=0}^4 A\left(\frac{j}{4}, \frac{k}{4}\right) + \sum_{j=1}^3 \sum_{k=1}^3 A\left(\frac{j}{4}, \frac{k}{4}\right) \right] \quad (19)$$

For the  $8 \times 8$  DCT case, the resulting global DC correction values for each 2-D AFT coefficient – based on Eqs. 16a and 16b – are provided in Table 1:

Table 1

1 k	0	1	2	3	4	5	6	7
0	0	$2 * E[A]$	$E[A]$	0	0	$-E[A]$	$-E[A]$	$-E[A]$
1	$2 * E[A]$	$-4 * E[A]$	$-2 * E[A]$	0	0	$2 * E[A]$	$2 * E[A]$	$2 * E[A]$
2	$E[A]$	$-2 * E[A]$	$-E[A]$	0	0	$E[A]$	$E[A]$	$E[A]$
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0
5	$-E[A]$	$2 * E[A]$	$E[A]$	0	0	$-E[A]$	$-E[A]$	$-E[A]$
6	$-E[A]$	$2 * E[A]$	$E[A]$	0	0	$-E[A]$	$-E[A]$	$-E[A]$
7	$-E[A]$	$2 * E[A]$	$E[A]$	0	0	$-E[A]$	$-E[A]$	$-E[A]$

10

The correction values for each 2-D AFT coefficient when there are non-zero column means and/or row means are provided in Table 2:

Table 2

1 k	1	2	3	4	5	6	7
1	$2x_{0,1} + 2x_{1,0}$	$2x_{0,2} + x_{1,0}$	$2x_{0,3}$	$2x_{0,4}$	$2x_{0,5} - x_{1,0}$	$2x_{0,6} - x_{1,0}$	$2x_{0,7} - x_{1,0}$
2	$x_{0,1} + 2x_{2,0}$	$x_{0,2} + x_{2,0}$	$x_{0,3}$	$x_{0,4}$	$x_{0,5} - x_{2,0}$	$x_{0,6} - x_{2,0}$	$x_{0,7} - x_{2,0}$
3	$2x_{3,0}$	$x_{3,0}$	0	0	$-x_{3,0}$	$-x_{3,0}$	$-x_{3,0}$

4	$2x_{4,0}$	$x_{4,0}$	0	0	$-x_{4,0}$	$-x_{4,0}$	$-x_{4,0}$
5	$-x_{0,1}+2x_{5,0}$	$-x_{0,2}+x_{5,0}$	$-x_{0,3}$	$-x_{0,4}$	$-x_{0,5}-x_{5,0}$	$-x_{0,6}-x_{5,0}$	$-x_{0,7}-x_{5,0}$
6	$-x_{0,1}+2x_{6,0}$	$-x_{0,2}+x_{6,0}$	$-x_{0,3}$	$-x_{0,4}$	$-x_{0,5}-x_{6,0}$	$-x_{0,6}-x_{6,0}$	$-x_{0,7}-x_{6,0}$
7	$-x_{0,1}+2x_{7,0}$	$-x_{0,2}+x_{7,0}$	$-x_{0,3}$	$-x_{0,4}$	$-x_{0,5}-x_{7,0}$	$-x_{0,6}-x_{7,0}$	$-x_{0,7}-x_{7,0}$

- Given the 2-D AFT coefficients  $x_{k,l}$  of the extended sub-image  $X$  and the corrected AFT coefficients  $A_c(k,l)$ , the DCT coefficients of the sub-image  $A$  can be calculated. The relations between the respective 8 x 8 point DCT coefficients
- 5 DCT( $k,l$ ) and the corresponding corrected 2-D AFT coefficients  $A_c(k,l)$  are provided in Table 3:

Table 3

1	0	1	2	3	4	5	6	7
k								
0	$A_c(8,8)*8$	$A_c(0,1)*4\sqrt{2}$	$A_c(0,2)*4\sqrt{2}$	$A_c(0,3)*4\sqrt{2}$	$A_c(0,4)*4\sqrt{2}$	$A_c(0,5)*4\sqrt{2}$	$A_c(0,6)*4\sqrt{2}$	$A_c(0,7)*4\sqrt{2}$
1	$A_c(1,0)*4\sqrt{2}$	$A_c(1,1)*4$	$A_c(1,2)*4$	$A_c(1,3)*4$	$A_c(1,4)*4$	$A_c(1,5)*4$	$A_c(1,6)*4$	$A_c(1,7)*4$
2	$A_c(2,0)*4\sqrt{2}$	$A_c(2,1)*4$	$A_c(2,2)*4$	$A_c(2,3)*4$	$A_c(2,4)*4$	$A_c(2,5)*4$	$A_c(2,6)*4$	$A_c(2,7)*4$
3	$A_c(3,0)*4\sqrt{2}$	$A_c(3,1)*4$	$A_c(3,2)*4$	$A_c(3,3)*4$	$A_c(3,4)*4$	$A_c(3,5)*4$	$A_c(3,6)*4$	$A_c(3,7)*4$
4	$A_c(4,0)*4\sqrt{2}$	$A_c(4,1)*4$	$A_c(4,2)*4$	$A_c(4,3)*4$	$A_c(4,4)*4$	$A_c(4,5)*4$	$A_c(4,6)*4$	$A_c(4,7)*4$
5	$A_c(5,0)*4\sqrt{2}$	$A_c(5,1)*4$	$A_c(5,2)*4$	$A_c(5,3)*4$	$A_c(5,4)*4$	$A_c(5,5)*4$	$A_c(5,6)*4$	$A_c(5,7)*4$
6	$A_c(6,0)*4\sqrt{2}$	$A_c(6,1)*4$	$A_c(6,2)*4$	$A_c(6,3)*4$	$A_c(6,4)*4$	$A_c(6,5)*4$	$A_c(6,6)*4$	$A_c(6,7)*4$
7	$A_c(7,0)*4\sqrt{2}$	$A_c(7,1)*4$	$A_c(7,2)*4$	$A_c(7,3)*4$	$A_c(7,4)*4$	$A_c(7,5)*4$	$A_c(7,6)*4$	$A_c(7,7)*4$

- 10 If the image signal being sampled has high spatial-frequency components that are not integer multiples of the unit spatial frequency, aliasing is likely to introduce a certain amount of error into the DCT coefficients computed with

the AFT algorithm. For example, as is illustrated in Fig. 2, there are sub-image boundaries 204 within the extended sub-image 202 derived from the original sub-image 102. At each of these boundaries there is likely to be a discontinuity in the first derivative of the pixel intensity. The discontinuities tend to increase as the input  
5 signal frequency approaches half the Nyquist sampling frequency. The discontinuities also tend to increase as the phase of the input signal approaches  $\pi/2$ . If substantial discontinuities are present, the extended sub-image 202 will have significant Fourier components at frequencies greater than half the Nyquist frequency. It is well known that if the Nyquist criterion is violated due to undersampling of an  
10 image signal or other signal, the high frequency harmonics — i.e., the components violating the Nyquist criterion — "fold back" to appear at frequencies below half the Nyquist frequency. An image extension, as such shown in Figure 2, does not lead to aliasing effects if the input signal is uniformly sampled at steps of 1/8th of the unit interval. However, due to the non-uniform placement of samples which have  
15 locations based on Farey fractions as is discussed above, aliasing errors may arise in DCT coefficients computed based on the AFT. The mean-square-error between uniformly sampled input signal values and an approximation of this signal — where the approximation is computed by taking the inverse DCT of the AFT-based DCT coefficients — provides an indication of the accuracy of the AFT-based procedure.  
20 The amount of error can be significant when processing image signals which have substantial high-frequency content. Exemplary results for mean-square error as a function of frequency are illustrated in Fig. 4, which plots, as a function of frequency, the mean-square error of the approximation signal obtained by taking the inverse DCT of the exemplary DCT coefficients derived by the above-described AFT technique.  
25 The illustrated results demonstrate that the error is greatest in the high-frequency components.

Error caused by undersampling not only directly affects the accuracy of filter outputs  $S(n,m)$  before any DC correction is applied, but also affects the accuracy of the DC correction itself.

30 An improved estimate for the mean value of the image may be obtained from the output of a filter,  $S$ , that averages a set of points taken at a spatial

frequency that is not expected to be present in the spectrum of the extended image  $X$  – P. Paparao, A. Ghosh, "An Improved Arithmetic Fourier Transform Algorithm," *SPIE Vol. 1347 Optical Information-Processing Systems and Architectures II* (1990). As a conclusion of the aforementioned paper, the increase in the order of the filter  $S$ , used to calculate the mean value, may improve the mean value estimate. Thus, the mean-square error should decrease when filters of order higher than 8 are used to estimate the mean value in the above-described  $8 \times 8$  DCT case. The density and the number of photosensitive elements that are averaged increases, when the higher order filters are used, so one should choose a filter with the highest realizable order, as limited by the fabrication technology. A particular fabrication technology limits the smallest distance between photosensitive elements, thus limiting the highest realizable filter order. In order to not significantly increase the number of photosensitive elements, the order of the filter should be divisible by at least one lower order. If the order of the filter is divisible by the lower order, the Farey fractions of the lower order filter would match to a subset of Farey fractions associated with a higher order filter, thus the number of additional photo-sensitive elements would not increase substantially. A typical example is the filter  $S(12,12)$ , where 12 is divisible by 2,3,4,6. A filter of order 12 requires no greater number of photosensitive elements than does a filter of order 8. However, for a  $12^{\text{th}}$ -order filter, the photosensitive elements are preferably more densely packed in certain parts of the sub-image, as is illustrated in Fig. 5. In general, in an  $N^{\text{th}}$ -order filter, sample locations may be placed at positions  $2j/N$ , where  $j=0,1,\dots,N/2$ . The estimated mean-square-error, where filters of order 12 are used to estimate the global and local mean values, is shown in Fig. 6.

Optionally, photosensitive elements located at the exact Farey fraction locations can be used to obtain the sample values for the high-order filter computations used to estimate the global and local DC values. Alternatively, or in addition, the sample values can be obtained by interpolation of neighboring samples using interpolation procedures discussed in further detail below. Furthermore, filters of order higher than 12 may be used to estimate the DC values. However, there is a tradeoff associated with using higher-order filters: such filters may entail an increase in the number of photosensitive elements and/or a decrease of the spacing between the

elements. Moreover, increasing the order of the filters beyond a value of 12 typically does not provide significant additional benefit. For example, Fig. 7 illustrates the mean-square error of a system which uses filters of order 16 to estimate the global and local DC values. A visual comparison of Figs. 6 and 7 reveals that the error is  
 5 approximately the same in both cases. It is therefore apparent that filters of order 12 provide a better tradeoff between the number of sample points (or pixel density) and the overall accuracy.

Aliasing errors in the non-DC-corrected filter outputs can be reduced by introducing additional pixels into the sensor array, provided that the fabrication  
 10 technology allows for a sufficiently dense pixel distribution. To correct for such aliasing, AFT coefficients of order higher than the equivalent uniform sampling frequency (i.e. coefficients of order higher than 8 for the 8x8 DCT case) can be used to correct the lower-order coefficients. The higher-order coefficients can be obtained directly from supplemental Farey-fraction-spaced sensors, interpolated from  
 15 neighboring pixels, or can be estimated as a fraction of the lower-order coefficients — methods which are described in further detail below.

By introducing additional pixels at the precise Farey fraction locations, it is possible to calculate the higher-order AFT coefficients exactly, which then may be used to correct the lower-order AFT coefficients. For example, if M is the number  
 20 of DCT coefficients, and the highest realizable order of the Farey fraction space is N, where  $N > M$  — i.e.  $N=9,10,11,12,\dots$  for  $M=8$ . First, the global and local DC corrections  $\Delta(k,l)$  and  $\Delta_{local}(k,l)$  are estimated using the highest order (N) filters as described above, and are added to the uncorrected AFT coefficients  $x_{k,l}$  as is indicated in Eqs. (18a) and (18b), above. The resulting DC-corrected AFT coefficients  $A_c(k,l)$ ,  
 25 where  $k,l = 0,1,2,\dots,N-1$ , are used to determine the aliasing-correction values:

$$\Delta_{alias}(k,l) = 0 \quad k = 0,1,\dots,2M-N; \quad l = 0,1,\dots,2M-N \quad (20a)$$

$$\Delta_{alias}(k,l) = -A_c(k,2M-l) \quad k = 0,1,\dots,2M-N; \quad l = 2M-N+1,\dots,M-1 \quad (20b)$$

$$\Delta_{alias}(k,l) = -A_c(2M-k,l) \quad k = 2M-N+1,\dots,M-1; \quad l = 0,1,\dots,2M-N \quad (20c)$$

$$\Delta_{\text{alias}}(k,l) = -A_c(k,2M-l) - A_c(2M-k,l) + A_c(2M-k,2M-l) \quad k=2M-N+1, \dots, M-1;$$

$$l=2M-N+1, \dots, M-1 \quad (20d)$$

The correction formulae in Eqs. (20a) - (20d) are valid when  $M$  is an even number — which is usually the case — and  $2M$  is greater than  $N$ . Aliasing-corrected 2-D AFT coefficients  $A_{cc}(k,l)$  can then be calculated by adding the above-

$$A_{cc}(k,l) = A_c(k,l) + \Delta_{\text{alias}}(k,l) \quad k=0,1, \dots, M-1; \quad l=0,1, \dots, M-1 \quad (21)$$

Fig. 8 illustrates the estimated mean-square error in an exemplary case in which higher Farey fraction samples are used to correct for aliasing. In the illustrated example, a Farey fraction sample space of order 12 (i.e.,  $N=12$ ) has been used to provide the pixel values, filters of order 12 have been used to estimate global and local DC values and higher-order AFT coefficients (coefficients of order 8,9,10, and 11) have been used to correct for aliasing as is discussed above with respect to Eqs. (20a) - (20d). In this example, the maximum estimated mean-square-error is at frequency (6.5,6.5) and is equal to 0.0273.

In Figs. 4 and 6-8, the estimated mean-square errors were derived by assuming for each frequency point  $(f_1, f_2)$  that the input image  $X$  is a 2-D cosine with frequency  $(f_1/2, f_2/2)$ . The 2-D AFT based 2-D DCT coefficients were calculated for such an input, and then an inverse 2-D DCT was calculated to obtain image  $Y$ . The mean-square error between image  $Y$  and  $X$  was calculated and assigned to the frequency point  $(f_1, f_2)$ .

The preferred number of image samples to be used for the AFT computation tends to increase substantially as the order of the Farey fraction space is increased. For example a total of 46 photosensitive elements per unit interval should be used when  $N=12$ . It may be impractical or expensive to fabricate image sensors with such a high pixel density, in which case the higher AFT coefficients are preferably estimated by interpolation of adjacent pixels. The Farey sampling points to be used for filters of order  $M, M+1, \dots, N-1$  can be interpolated either from the available set of samples or from the set of samples being processed by a particular filter,

preferably the highest order filter N (the 12th-order filter in the example given above). In any case, an exemplary interpolation system is discussed in further detail below.

In an additional method for calculating higher-order 2-D AFT coefficients (e.g., coefficients of order 8, 9, 10, and 11), the higher-order coefficients are calculated as a fraction of the neighboring higher-order coefficients. Specifically, one or more higher-order coefficients are first calculated using exact Farey sampling points, and the other higher-order coefficients can be estimated from these exact values as follows. Assuming that the image A is band-limited and has no frequency components beyond half the Nyquist frequency, the correlation between respective neighboring, higher-order Fourier series coefficients is typically quite high. Moreover, simulations have shown that even Fourier series coefficients (coefficients 8,10,12 in our example) tend to be highly correlated with each other, and similarly, odd Fourier series coefficients (coefficients 9 and 11 in our example) tend to be highly correlated with each other. Accordingly, if one even, higher-order Fourier coefficient is known, the other even, higher-order coefficients can be estimated. Similarly, if one odd, higher-order Fourier coefficient is known, the other odd, higher-order coefficients can be estimated. For example, if a Farey fraction space of order 7 and filters of order 12 are used, filters of order 9 can be used to estimate the odd higher-order coefficients, and filters of order 12 can be used to estimate the even higher-order coefficients. An exemplary sample space suitable for such estimations (for aliasing-correction) is illustrated in Fig. 9, where locations of the photosensitive elements are defined as Farey fractions  $2j/n$ ;  $j=0,1,2,\dots,n-1$  and  $n=1,2,3,4,5,6,7,9$ , and 12.

In addition, a single system can combine the above-described techniques of: (a) adding sensors at higher-order Farey fraction locations, and (b) interpolating the values from existing sensors to estimate the values of the incoming signal at the appropriate higher-order locations. For example, as is illustrated in Fig. 9, if a desired higher-order pixel location 906 is quite close to a lower-order pixel location 904, and there is a sensor at the lower-order location 904, it may be preferable to compute an estimated value for the higher-order pixel 906 by interpolation, rather than by placing a sensor at the higher-order location 906.

However, if a desired higher-order location 910 is farther away from the nearest lower-order pixels 908 and 912, it may be preferable to add an extra sensor to the sensor array in the higher-order location 910.

Fig. 16 provides an overview of an exemplary procedure for image sensing and processing in accordance with the present invention. Pixel values 1602 are processed to calculate the filters  $S(n,m)$  according to Eq.(14) above (step 1604). A set of uncorrected AFT coefficients  $x_{k,l}$  are computed based upon the filter values  $S(n,m)$  (step 1606). If the entire image and the respective rows and columns have no non-zero DC components, no mean value correction is required (step 1608). The AFT coefficients  $x_{k,l}$  are therefore power-normalized — as is illustrated above in Eqs. (15e)-(15h) — to derive the DCT coefficients 1618 (step 1616). If, however, a mean value correction is appropriate (step 1608) the mean value correction amounts are computed (step 1610) and used to correct the AFT coefficients  $x_{k,l}$  for deriving corrected coefficients  $A_c(k,l)$  (step 1612). If no aliasing correction is required (step 1614), the procedure continues to step 1616. However, if aliasing correction is appropriate (step 1614), the aliasing corrections are computed as discussed above (step 1620), and used to further correct the DC-corrected AFT coefficients  $A_c(k,l)$  for deriving alias-corrected coefficients  $A_{cc}(k,l)$  (step 1622). The DCT coefficients 1618 are then calculated based on the alias-corrected AFT coefficients  $A_{cc}(k,l)$  (step 1616).

As is discussed above, interpolation of measurements from neighboring sensors in a sensor array can be useful for estimating the value of a pixel adjacent to the locations of the sensors. For example, referring to the unit area 300 illustrated in Fig. 3, if the AFT method of the present invention is to be used with a conventional sensor array having sensors located in uniformly spaced positions 398, interpolation can be used to estimate the values of the image at the Farey-fraction-based locations 348. If the computation is being performed by a digital signal processor such as the digital filter 1502 illustrated in Fig. 15, the computation of the value at a particular Farey fraction location 345 can, for example, be performed by computing an average of the respective values generated by the sensors located at the nearest uniformly spaced locations 394, 395, 396, and 397.



Fig. 13 illustrates an exemplary procedure for deriving AFT coefficients using interpolated pixel values. In the illustrated procedure, an incoming image signal is received by a sensor array (step 1302). The sensor array can, for example, be a conventional array having sensors with uniformly distributed spatial locations. The incoming signal is detected by the sensors of the array to generate a plurality of sensor signals (1304). The sensor signals are received by an interpolation circuit (step 1306) which interpolates the sensor signals (step 1308) — e.g., by averaging the signals — to generate a set of interpolated signals which represent the pixel values at locations defined by Farey fractions as is discussed above. The interpolated signals are received by a filter arrangement such as the analog filter 1022 illustrated in Fig. 10 or the digital filter 1502 illustrated in Fig. 15 (step 1310). The filter 1022 or 1502 derives respective weighted sums of respective sets of interpolated signals to generate respective filtered signals (step 1316). For example, a weighted sum of a first set of interpolated signals is derived to generate a first filtered signal (step 1312), and a weighted sum of a second set of interpolated signals is derived to generate a second filtered signal (step 1314). In the case of an analog filter arrangement 1022, the weighted sums derived in steps 1312 and 1314 can be produced in accordance with the procedure illustrated in Fig. 14. In the illustrated filtering procedure 1312 or 1314, the interpolated signals from a particular rows and columns are amplified with the appropriate gains to generate respective amplified signals (step 1408). For example, a first interpolated signal is amplified with a first gain to generate a first amplified signal (step 1402), a second interpolated signal is amplified with a second gain to generate a second amplified signal (step 1404), etc. The resulting amplified signals are integrated to generate the filtered signal (step 1406).

Once the respective filter outputs  $S(m,n)$  are derived, the 2-D AFT coefficients are derived (step 1112). To derive the AFT coefficients (step 1112), the filter outputs are weighted using appropriate values of a Mobius function as is described above with respect to Eqs. (15a)-(15d) above (step 1114), and the resulting weighted signals are added/summed in accordance with Eqs. (15a)-(15d) (step 1116).

Further improvement of computational efficiency can be achieved by using an analog circuit to perform the aforementioned interpolation. Fig. 18 illustrates an exemplary analog interpolation circuit 1804 for interpolating pixel values from sensors 1806 of a sensor array portion 1802 to derive additional pixels 1808, 1810, and 1812 (pixels of the row 1814 and column 1816) for use in an AFT computation in accordance with the present invention. To interpolate pixels 1808 of the row 1814, the pixels 1826 of the rows 1818 and 1820 are used. Similarly, to interpolate the pixels 1810 of the column 1816, the pixels 1828 of the columns 1822 and 1824 are used. Although the pixels of interest are not necessarily equidistant from their neighboring pixels, they can be approximated as equidistant, which results in a 0.5% error. Each interpolated pixel value is therefore approximated as the average value of the two neighboring pixel values. A special case is the pixel 1812 at the location where row 1814 and column 1816 intersect. This pixel value will be interpolated as an average value of four neighboring pixels (pixel values 1830 at the intersections (1818,1822), (1818,1824), (1820,1822), and (1820,1824)). Assuming that the pixels of interest are equidistant from their nearest neighbors allows a minimum number of sampling capacitors to be used.

An exemplary timing cycle for calculating the filter  $S(3,12)$  using the interpolation circuit 1804 is provided below:

1.  $n=3, m=12,$
2.  $\Phi_{int}^3=1, \Phi_{int}^i=0$ , where  $i = 1,2,4,5,6,7,12,$
3. Select Column 0
4.  $\Phi_{s2}^1=1, \Phi_{s2}^8=1$ , other  $\Phi_{sj}^i=0$
5. Transfer charge to integrator 1832,  $\Phi_t=1, \Phi_{sj}^i=0$
6. Select Column 1/6
7.  $\Phi_{s1}^1=1, \Phi_{s1}^8=1$ , other  $\Phi_{sj}^i=0$
8. Transfer charge to integrator 1832,  $\Phi_t=1, \Phi_{sj}^i=0$
9. Select Column 1/3

10.  $\Phi_{s1}^1=1$ ,  $\Phi_{s1}^8=1$ , other  $\Phi_{sj}^i=0$
11. Transfer charge to integrator 1832,  $\Phi_t=1$ ,  $\Phi_{sj}^i=0$
12. Select Column 1/2
13.  $\Phi_{s1}^1=1$ ,  $\Phi_{s1}^8=1$ , other  $\Phi_{sj}^i=0$
- 5 14. Transfer charge to integrator 1832,  $\Phi_t=1$ ,  $\Phi_{sj}^i=0$
15. Select Column 2/3
16.  $\Phi_{s1}^1=1$ ,  $\Phi_{s1}^8=1$ , other  $\Phi_{sj}^i=0$
17. Transfer charge to integrator 1832,  $\Phi_t=1$ ,  $\Phi_{sj}^i=0$
18. Select Column 4/5 – interpolation column
- 10 19.  $\Phi_{s2}^1=1$ ,  $\Phi_{s2}^8=1$ , other  $\Phi_{sj}^i=0$  – note that values in column 4/5 are sampled  
with gain 2 instead of gain 4
20. Transfer charge to integrator 1832,  $\Phi_t=1$ ,  $\Phi_{sj}^i=0$
21. Select Column 6/7 – interpolation column
22.  $\Phi_{s2}^1=1$ ,  $\Phi_{s2}^8=1$ , other  $\Phi_{sj}^i=0$  – note that values in column 6/7 are sampled  
15 with the gain 2 instead of gain 4
23. Transfer charge to integrator 1832,  $\Phi_t=1$ ,  $\Phi_{sj}^i=0$
24. Sample the integrators output,  $\Phi_{s3}=1$
25.  $\Phi_{amp}^{12}=1$ ,  $\Phi_{amp}^i=0$ , where  $i = 1,2,3,4,5,6,7$ ,
26. Transfer charge to amplifier 1834,  $\Phi_{s3}=0$ ,  $\Phi_{s3}=1$
- 20 27. Perform AD conversion using ADC 1836 and store the digital value S(3,12)  
in RAM 1838
28. Reset the integrator and amplifier

Table 4 presents a comparison of the computational efficiencies of several different methods for computing a 1-D, 8-point DCT, including the AFT method of the present invention. The comparison is expressed in terms of the respective numbers of various types of operations used to compute the 1-D DCT:

Table 4

	Conventional DCT	Fast DCT in [Chen 1977]	Fast DCT in [Narasimha 1978]	AFT
Additions	56	26	NA	33
Multiplications	42	16	18	7
Total Operations (8 pt) 1-D DCT	1820	282	350	82
Total Operations (8x8 ) 2-D DCT Normalized to AFT	493	12	18	1

It can be seen from Table 4 that, in terms of the total number of operations, the AFT method of the present invention is approximately 3.4 times as efficient as the most efficient prior art method for computing a 1-D DCT.

Furthermore, because the number of total operations in the 2-D case is approximately proportional to the square of the number of computations in the 1-D case, the AFT method of the present invention is approximately 12 times as efficient as the most efficient prior art method for computing a 2-D DCT. In addition, because the multiplications in the AFT computation comprise pre-scaling of the respective pixel intensities by integer values, these multiplications can be readily implemented using analog circuits such as the filter 1022 illustrated in Fig. 10. By effectively eliminating most of the digital multiplications, such an analog filter 1022 allows the AFT system of the present invention to use 73 times fewer computations than the most efficient prior art system.

Although the present invention has been described in connection with specific exemplary embodiments, it should be understood that various changes, substitutions, and alterations can be made to the disclosed embodiments without

departing from the spirit and scope of the invention as set forth in the appended claims.

### Appendix A

This Appendix provides a proof of the following relation:

$$S(n, m, p_{ref}, q_{ref}) = \sum_{\substack{j \text{ such that} \\ n|j}} \sum_{\substack{k \text{ such that} \\ m|k}} a_{j,k}(p_{ref}, q_{ref}) \quad (\text{A-1})$$

The outputs of the filters are as follows:

$$5 \quad S(n, m, p_{ref}, q_{ref}) = \frac{1}{n} \frac{1}{m} \sum_{j=1}^{n-1} \sum_{k=1}^{m-1} A(p_{ref} - \frac{j}{n}, q_{ref} - \frac{k}{m}) \quad (\text{A-2})$$

and the Fourier Series extension of the image A is provided by Eqs. (6) and (7), which are reproduced as follows:

$$10 \quad A(p_{ref}, q_{ref}) = \sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{k,l}(p_{ref}, q_{ref}) \quad (\text{A-3a})$$

$$a_{k,l}(p_{ref}, q_{ref}) = A_{k,l} \cos(2\pi k p_{ref} + \theta_k) \cos(2\pi l q_{ref} + \theta_l) \quad (\text{A-3b})$$

Thus the filters' output formula (Eq. (A-2)) can be written as follows:

$$S(n, m, p_{ref}, q_{ref}) = \frac{1}{n} \frac{1}{m} \sum_{j=1}^{n-1} \sum_{k=1}^{m-1} \sum_{o=1}^{\infty} \sum_{r=1}^{\infty} A_{o,p} \operatorname{Re} \left\{ e^{j \left( 2\pi o \left( p_{ref} - \frac{j}{n} \right) + \theta_o \right)} \right\} \cdot \operatorname{Re} \left\{ e^{j \left( 2\pi r \left( q_{ref} - \frac{k}{m} \right) + \theta_r \right)} \right\} \quad (\text{A-4})$$

15 Rearranging the summation order, Equation (A-4) can be written as in (A-5)

$$S(n, m, p_{ref}, q_{ref}) = \sum_{o=1}^{\infty} \sum_{r=1}^{\infty} a_{o,r}(p_{ref}, q_{ref}) \cdot \frac{1}{n} \sum_{j=1}^{n-1} \operatorname{Re} \left\{ e^{-j \left( 2\pi o \frac{j}{n} \right)} \right\} \cdot \frac{1}{m} \sum_{k=1}^{m-1} \operatorname{Re} \left\{ e^{-j \left( 2\pi r \frac{k}{m} \right)} \right\} \quad (\text{A-5})$$

Having the relation (A-6), the filters' outputs become as in (A-7)

$$\frac{1}{n} \sum_{j=1}^{n-1} \operatorname{Re} \left\{ e^{-j \left( 2\pi o \frac{j}{n} \right)} \right\} = \begin{cases} 1 & n | o \\ 0 & \text{otherwise} \end{cases} \quad (\text{A-6})$$

$$20 \quad S(n, m, p_{ref}, q_{ref}) = \sum_{o=1}^{\infty} \sum_{r=1}^{\infty} a_{on,rm}(p_{ref}, q_{ref}) = \sum_{\substack{j \text{ such that} \\ n|j}} \sum_{\substack{k \text{ such that} \\ m|k}} a_{j,k}(p_{ref}, q_{ref}) \quad (\text{A-7})$$

## Appendix B

This Appendix provides a proof of the following relation:

$$a_{k,l}(p_{ref}, q_{ref}) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu_2(m, n) \cdot S(mk, nl, p_{ref}, q_{ref}) \quad \text{for } k, l = 1, 2, \dots, N \quad (\text{B-1})$$

5 The Kroneker function is defined as follows:

$$\delta(n, m) = 1 \quad \text{for } n = m, \quad (\text{B-2a})$$

$$\delta(n, m) = 0 \quad \text{elsewhere,} \quad (\text{B-2b})$$

The Mobius function  $\mu_1$  and Kroneker function  $\delta$  are related as follows:

$$\delta(m, n) = \sum_{d|(m/n)} \mu_1(d) = \begin{cases} 1, & \text{if } \frac{m}{n} = 1 \\ 0, & \text{otherwise} \end{cases} \quad (\text{B-3})$$

10 The values of  $m$  and  $n$  are positive integers and summations are carried out over all positive integer values of  $d$  that exactly divide the positive integer  $m/n$ .

In order to prove the relation in Eq. (B-1), Eqs. (B-3) and (9) can be used to derive the following relations:

$$\begin{aligned} \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu_2(m, n) \cdot S(mk, nl, p_{ref}, q_{ref}) &= \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \mu_1(m) \mu_1(n) \sum_{p=1}^{\infty} \sum_{q=1}^{\infty} a_{mkpnlq}(p_{ref}, q_{ref}) \\ &= \sum_{w=1}^{\infty} \sum_{v=1}^{\infty} a_{w,v}(p_{ref}, q_{ref}) \left( \sum_{m|w/k} \mu_1(m) \right) \left( \sum_{n|v/l} \mu_1(n) \right) \\ &= \sum_{w=1}^{\infty} \sum_{v=1}^{\infty} a_{w,v}(p_{ref}, q_{ref}) \delta(w, k) \cdot \delta(v, l) \\ &= a_{kl}(p_{ref}, q_{ref}) \end{aligned} \quad (\text{B-4})$$

## Appendix C

### 2D DCT and 2DAFT coefficients equality

Image X is the extended version of the unit area sub-image A (as shown in Figure 1). According to the two-dimensional case of the Nyquist reconstruction formula, the continuous image X can be represented by its samples as follows.

$$X(p, q) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} X\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \cdot \text{sinc}\left(\frac{p - \left(n + \frac{1}{2}\right)T}{T}\right) \cdot \text{sinc}\left(\frac{q - \left(m + \frac{1}{2}\right)T}{T}\right) \quad (\text{C-1})$$

Without losing generality we will assume that the sampling period  $T$  is the same in both dimensions and equals  $1/8$ . As a result, there are  $16 \times 16$  samples. Let us assume that the image  $X$  is periodic with period  $2 \times 2$  units. Thus, Equation (C-1) can be written as follows:

$$X(p, q) = \sum_{n=0}^{15} \sum_{m=0}^{15} X\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \cdot \sum_{k=-\infty}^{\infty} \text{sinc}\left(\frac{p - (16k + n + \frac{1}{2})T}{T}\right) \cdot \sum_{l=-\infty}^{\infty} \text{sinc}\left(\frac{q - (16l + m + \frac{1}{2})T}{T}\right) \quad (\text{C-2})$$

It can be shown using the inverse Fourier transform and the dual form of the Poisson formula that the summation of the sinc-functions is equal to the right side of Eq. (C-3):

$$\sum_{k=-\infty}^{\infty} \text{sinc}\left(\frac{p - (16 \cdot k + n + \frac{1}{2})T}{T}\right) = \frac{1}{16} \sum_{k=-8}^8 e^{j \frac{2\pi k}{16T} \left(p - \frac{T}{2} - nT\right)} \quad (\text{C-3})$$

Based on Eq. (C-3), Eq. (C-2) can be written as follows:

$$X(p, q) = \sum_{n=0}^{15} \sum_{m=0}^{15} X\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \cdot \frac{1}{16} \sum_{k=-8}^8 e^{j \frac{2\pi k}{16T} \left(p - \frac{T}{2} - nT\right)} \cdot \frac{1}{16} \sum_{l=-8}^8 e^{j \frac{2\pi l}{16T} \left(q - \frac{T}{2} - mT\right)} \quad (\text{C-4})$$



Because  $X(p, q)$  is the extended version of the image  $A(p, q)$ , as expressed in (C-5), Eq. (C-4) can be rearranged into Eq. (C-6)

$$X(p, q) = \begin{cases} A(p, q) & 0 \leq p < 1, 0 \leq q < 1, \\ A(2-p, q) & 1 \leq p < 2, 0 \leq q < 1, \\ A(p, 2-q) & 0 \leq p < 1, 1 \leq q < 2, \\ A(2-p, 2-q) & 1 \leq p < 2, 1 \leq q < 2 \end{cases} \quad (C-5)$$

$$5 \quad X(p, q) = \sum_{n=0}^7 \sum_{m=0}^7 A\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \cdot \frac{1}{8} \sum_{k=-8}^8 e^{j \frac{2\pi k}{16T} p} \cos(k\pi) \cos\left(\frac{\pi \cdot k}{16} (15 - 2n)\right) \\ \cdot \frac{1}{8} \sum_{l=-8}^8 e^{j \frac{2\pi l}{16T} q} \cos(l\pi) \cos\left(\frac{\pi \cdot l}{16} (15 - 2m)\right) \quad (C-6)$$

The product term of the cosine functions is as follows:

$$\cos(k\pi) \cos\left(\frac{\pi \cdot k}{16} (15 - 2n)\right) = \cos\left(\frac{\pi \cdot k \cdot (2n + 1)}{16}\right) \quad (C-7)$$

- 10 Replacing the product terms with (C-7) and rearranging the order of the summations Equation (C-6) becomes the following:

$$X(p, q) = \sum_{k=-8}^8 \sum_{l=-8}^8 e^{j \cdot \pi \cdot k \cdot p} \cdot e^{j \cdot \pi \cdot l \cdot q} \cdot \frac{1}{64} \sum_{n=0}^7 \sum_{m=0}^7 A\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \cdot \\ \cos\left(\frac{\pi \cdot k \cdot (2n + 1)}{16}\right) \cdot \cos\left(\frac{\pi \cdot l \cdot (2m + 1)}{16}\right) \quad (C-8)$$

- 15 From Eq. (C-8) it can be seen that the (n,m) summation term does not depend on the sign of k and l. Also, according to the definition of the two-dimensional DCT given in (C-10), Eq. (C-8) can be written as follows:

$$X(p, q) = \frac{1}{64} \sum_{n=0}^7 \sum_{m=0}^7 A\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] + \\ + \sum_{k=1}^8 \left( \frac{1}{32} \sum_{n=0}^7 \sum_{m=0}^7 A\left[\left(n + \frac{1}{2}\right)T, \left(m + \frac{1}{2}\right)T\right] \right) \cdot \cos\left(\frac{\pi \cdot k \cdot (2n + 1)}{16}\right) \cdot \cos(k \cdot \pi \cdot p) +$$

$$\begin{aligned}
& + \sum_{l=1}^8 \left( \frac{1}{32} \sum_{n=0}^7 \sum_{m=0}^7 A \left[ \left( n + \frac{1}{2} \right) T, \left( m + \frac{1}{2} \right) T \right] \right) \cdot \cos \left( \frac{\pi \cdot l \cdot (2m+1)}{16} \right) \cdot \cos(l \cdot \pi \cdot q) + \\
& + \sum_{k=1}^8 \sum_{l=1}^8 \left( \frac{1}{64} \sum_{n=0}^7 \sum_{m=0}^7 A \left[ \left( n + \frac{1}{2} \right) T, \left( m + \frac{1}{2} \right) T \right] \cdot \cos \left( \frac{\pi \cdot k \cdot (2n+1)}{16} \right) \cdot \cos \left( \frac{\pi \cdot l \cdot (2m+1)}{16} \right) \right) \cdot \\
& \cdot \cos(k \cdot \pi \cdot p) \cos(l \cdot \pi \cdot q)
\end{aligned} \tag{C-9}$$

5

The definition of the two-dimensional DCT is as follows:

$$DCT\{A\}(k,l) = \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} \alpha(k) \cdot \alpha(l) \cdot A \left[ \left( n + \frac{1}{2} \right) T, \left( m + \frac{1}{2} \right) T \right] \cdot \cos \left( \frac{\pi \cdot k \cdot (2n+1)}{2N} \right) \cdot \cos \left( \frac{\pi \cdot l \cdot (2m+1)}{2N} \right)$$

(C-10)

where:

$$10 \quad \alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \sqrt{\frac{2}{N}}, \quad k = 1, 2, 3, \dots, N-1$$

Eq. (C-9) can therefore be written as follows:

$$\begin{aligned}
X(p, q) &= \frac{DCT\{A\}(0,0)}{8} + \sum_{k=1}^8 \frac{DCT\{A\}(k,0)}{4\sqrt{2}} \cos(k \cdot \pi \cdot p) + \\
15 \quad & + \sum_{l=1}^8 \frac{DCT\{A\}(0,l)}{4\sqrt{2}} \cos(l \cdot \pi \cdot q) + \sum_{k=1}^8 \sum_{l=1}^8 \frac{DCT\{A\}(k,l)}{4} \cos(k \cdot \pi \cdot p) \cos(l \cdot \pi \cdot q) \tag{C-11}
\end{aligned}$$

In addition, the extended image  $X(p, q)$  can be represented by its two-dimensional Fourier series:

$$X(p, q) = E[X] + \sum_{k=1}^8 \overline{x_{k,0}} \cos(k \cdot \pi \cdot p) +$$

$$+ \sum_{l=1}^8 \overline{x_{0,l}} \cos(1 \cdot \pi \cdot q) + \sum_{k=1}^8 \sum_{l=1}^8 x_{k,l} \cos(k \cdot \pi \cdot p) \cos(1 \cdot \pi \cdot q) \quad (C-12)$$

where  $x_{k,l}$  ( $k, l=1, 2, \dots, 8$ ) are 2DAFT coefficients of the extended image X. The second and third terms of Eq. (C-12) are due to the presence of local row and column non-zero mean values. The coefficients inside the second term are calculated as the 1D AFT of the row-means; and the coefficients inside the third term are calculated as the 1D AFT of the column-means.

Having representations (C-11) and (C-12) of the image  $X(t, \tau)$ , and having orthogonal cosine functions in both formulae, we can conclude that the 2DAFT and DCT coefficients are equal except for a constant multiplicative factor in each DCT coefficient:

$$DCT\{A\}(0,0) = 8 * E[A],$$

$$DCT\{A\}(k,0) = 4\sqrt{2} * \overline{x_{k,0}} \quad k = 1, 2, 3, \dots, N-1$$

$$DCT\{A\}(0,l) = 4\sqrt{2} * \overline{x_{0,l}} \quad l = 1, 2, 3, \dots, N-1$$

$$DCT\{A\}(k,l) = 4 * x_{k,l} \quad k = 1, 2, 3, \dots, N-1, \quad l = 1, 2, 3, \dots, N-1 \quad (C-13)$$